JAVA.LANG.STRINGBUFFER CLASS

Introduction

The **java.lang.StringBuffer** class is a thread-safe, mutable sequence of characters. Following are the important points about StringBuffer:

- A string buffer is like a String, but can be modified.
- It contains some particular sequence of characters, but the length and content of the sequence can be changed through certain method calls.
- They are safe for use by multiple threads.
- Every string buffer has a capacity.

Class declaration

Following is the declaration for **java.lang.StringBuffer** class:

```
public final class StringBuffer
   extends Object
      implements Serializable, CharSequence
```

Class constructors

S.N.	Constructor & Description
1	**StringBuffer()** This constructs a string buffer with no characters in it and an initial capacity of 16 characters.
2	**StringBuffer(CharSequence seq)** This constructs a string buffer that contains the same characters as the specified CharSequence.
3	**StringBuffer(int capacity)** This constructs a string buffer with no characters in it and the specified initial capacity.
4	**StringBuffer(String str)** This constructs a string buffer initialized to the contents of the specified string.

Class methods

S.N.	Method & Description
1	StringBuffer append(boolean b) This method appends the string representation of the boolean argument to the sequence
2	StringBuffer append(char c) This method appends the string representation of the char argument to this sequence.

3	StringBuffer append(char[] str) This method appends the string representation of the char array argument to this sequence.
4	StringBuffer append(char[] str, int offset, int len) This method appends the string representation of a subarray of the char array argument to this sequence.
5	StringBuffer append(CharSequence s) This method appends the specified CharSequence to this sequence.
6	StringBuffer append(CharSequence s, int start, int end) This method appends a subsequence of the specified CharSequence to this sequence.
7	StringBuffer append(double d) This method appends the string representation of the double argument to this sequence.
8	StringBuffer append(float f) This method appends the string representation of the float argument to this sequence.
9	StringBuffer append(int i) This method appends the string representation of the int argument to this sequence.
10	StringBuffer append(long lng) This method appends the string representation of the long argument to this sequence.
11	StringBuffer append(Object obj) This method appends the string representation of the Object argument.
12	StringBuffer append(String str) This method appends the specified string to this character sequence.
13	StringBuffer append(StringBuffer sb) This method appends the specified StringBuffer to this sequence.
14	StringBuffer appendCodePoint(int codePoint) This method appends the string representation of the codePoint argument to this sequence.
15	int capacity() This method returns the current capacity.
16	char charAt(int index) This method returns the char value in this sequence at the specified index.
17	int codePointAt(int index) This method returns the character (Unicode code point) at the specified index
18	int codePointBefore(int index) This method returns the character (Unicode code point) before the specified index
19	int codePointCount(int beginIndex, int endIndex) This method returns the number of Unicode code points in the specified text range of this sequence
20	StringBuffer delete(int start, int end) This method removes the characters in a substring of this sequence.
21	StringBuffer deleteCharAt(int index) This method removes the char at the specified position in this sequence

22	void ensureCapacity(int minimumCapacity) This method ensures that the capacity is at least equal to the specified minimum.
23	void getChars(int srcBegin, int srcEnd, char[] dst, int dstBegin) This method characters are copied from this sequence into the destination character array dst.
24	int indexOf(String str) This method returns the index within this string of the first occurrence of the specified substring.
25	int indexOf(String str, int fromIndex) This method returns the index within this string of the first occurrence of the specified substring, starting at the specified index.
26	StringBuffer insert(int offset, boolean b) This method inserts the string representation of the boolean argument into this sequence.
27	StringBuffer insert(int offset, char c) This method inserts the string representation of the char argument into this sequence.
28	StringBuffer insert(int offset, char[] str) This method inserts the string representation of the char array argument into this sequence.
29	StringBuffer insert(int index, char[] str, int offset, int len) This method inserts the string representation of a subarray of the str array argument into this sequence.
30	StringBuffer insert(int dstOffset, CharSequence s) This method inserts the specified CharSequence into this sequence.
31	StringBuffer insert(int dstOffset, CharSequence s, int start, int end) This method inserts a subsequence of the specified CharSequence into this sequence.
32	StringBuffer insert(int offset, double d) This method inserts the string representation of the double argument into this sequence.
33	StringBuffer insert(int offset, float f) This method inserts the string representation of the float argument into this sequence.
34	StringBuffer insert(int offset, int i) This method inserts the string representation of the second int argument into this sequence.
35	StringBuffer insert(int offset, long l) This method inserts the string representation of the long argument into this sequence.
36	StringBuffer insert(int offset, Object obj) This method inserts the string representation of the Object argument into this character sequence.
37	StringBuffer insert(int offset, String str) This method inserts the string into this character sequence.
38	int lastIndexOf(String str) This method returns the index within this string of the rightmost occurrence of the specified substring.
39	int lastIndexOf(String str, int fromIndex) This method returns the index within this string of the last occurrence of the specified substring.
40	int length() This method returns the length (character count).

41	int offsetByCodePoints(int index, int codePointOffset) This method returns the index within this sequence that is offset from the given index by codePointOffset code points.
42	StringBuffer replace(int start, int end, String str) This method replaces the characters in a substring of this sequence with characters in the specified String.
43	StringBuffer reverse() This method causes this character sequence to be replaced by the reverse of the sequence.
44	void setCharAt(int index, char ch) The character at the specified index is set to ch.
45	void setLength(int newLength) This method sets the length of the character sequence.
46	CharSequence subSequence(int start, int end) This method returns a new character sequence that is a subsequence of this sequence.
47	String substring(int start) This method returns a new String that contains a subsequence of characters currently contained in this character sequence
48	String substring(int start, int end) This method returns a new String that contains a subsequence of characters currently contained in this sequence.
49	String toString() This method returns a string representing the data in this sequence.
50	void trimToSize() This method attempts to reduce storage used for the character sequence.

Methods inherited

This class inherits methods from the following classes:

- java.lang.Object